WHAT IS CLAIMED IS

1.    A method comprising:

assigning a logical level number to a symbol in a key comprising a string of symbols; and

storing an entry for the key in a level of nodes of a multiway radix tree based at least in part on the logical level number instead of on a path between nodes representing every symbol in the key.

2.    The method of claim 1, wherein the data structure is comprised of levels of nodes arranged according to ascending magnitude of logical level numbers assigned to the levels of nodes.

3.    The method of claim 2, wherein a minimum number of symbols of the key are used to differentiate the key from other keys stored on the tree, and the data structure has only levels of nodes for the symbols of the key that are used to differentiate the key from other keys stored on the tree.

4.    The method of claim 2, wherein a search for the key uses the logical level number corresponding to a length of the key to find the level of nodes storing the key.

5.    A method comprising:

creating a root node and first-level nodes of a tree for storing keys in a memory, each key having a string of symbols;

creating a pointer from the root node to a node in the first-level of nodes corresponding to the first symbol in a key;

creating a second level of nodes to store an entry for the key in a node corresponding to the last symbol in the key; and

creating a pointer from the node in the first-level of nodes corresponding to the first symbol in the key to the node in the second level of nodes corresponding to the last symbol in the key.

6.      The method of claim 5, further comprising assigning a first logical level value of one, corresponding to the first symbol in the key, to the logical level number for the first level of nodes; and

assigning a second logical level value of n to the logical level number for the second level of nodes corresponding to the last symbol in the key, wherein n equals the number of symbols in the key.

7.      The method of claim 6, wherein a new level of nodes to store an entry for a new key in a node corresponding to the last symbol in the new key is added to and/or inserted between the existing levels of nodes in the tree based on ordering all the levels of nodes in the tree in sequence according to ascending logical level numbers.

8.      The method of claim 7, further comprising rearranging pointers that exist between a parent level of nodes and a child level of nodes when a level of nodes having a logical level number between the logical level numbers of the parent level of nodes and the child level of nodes is inserted between the parent level of nodes and the child level of nodes.

9.      The method of claim 8, wherein the rearranging includes maintaining a logical path between nodes representing symbols in a key as the number of logical levels between nodes representing a first symbol in the key and a last symbol in the key changes.

10.     A method, comprising:

a)      ascertaining a logical level number that corresponds to the length of a key comprising a string of n symbols in a table of stored key entries organized as a tree of nodes;

b)      searching a root node in the tree for a pointer to a node corresponding to a first symbol in the key;

19

c)      searching the node corresponding to the first symbol in the key for a pointer to a next level of nodes;

d)      ascertaining whether the next level of nodes is assigned the logical level number that corresponds to the length of the key;

e)      searching for an entry for the key on the next level of nodes if the next level of nodes is assigned the logical level number;

f)      searching for a pointer on the next level of nodes to a subsequent next level of nodes if the next level of nodes is not assigned the logical level number that corresponds to the length of the key; and

g)      repeating d-f until one of the key entry is found, there are no more pointers to subsequent levels of nodes, and a number of levels of nodes corresponding to a number of all the symbols in the key have been searched.


11.      The method of claim 10, further comprising inserting a new level of nodes to store the key if the key is not found, comprising:

following a first pointer from the root node to the node corresponding to the first symbol in the key;

searching the node corresponding to the first symbol in the key for a second pointer to a next level of nodes;

if the second pointer does not exist, then creating a next level of nodes, storing an entry for the key in a node corresponding to the last symbol of the key, and assigning logical level number corresponding to a length of the key to the level of nodes in which the key is stored;

if the second pointer exists and points to a level of nodes having a logical level number corresponding to the length of the key, then storing an entry for the key in a node corresponding to the last symbol of the key; and

if the second pointer exists and points to a level of nodes not having a logical level number corresponding to a length of the key, then inserting a new level of nodes to store an entry for the key, and inserting the new level according to ascending magnitude of the logical level numbers of all the levels of nodes in the tree, starting at the root node.

12.     An apparatus, comprising:

a register to store an entry for a key having a string of symbols in a tree data structure representing a table; and

a node generator to add a node to the tree to store the key entry based on the last symbol of the key, wherein the node is assigned a logical level number corresponding to the length of the key; and

wherein the logical level number of the node storing the key is used to find the key instead of a path between nodes corresponding to each symbol in the key.

13.     The apparatus of claim 12, further comprising:

a receiver to receive keys; and

an inserter to insert an entry for each key to be stored into the node created by the node generator.

14.     The apparatus of claim 12, further comprising:

a finder to:

follow a pointer to a node corresponding to the first symbol of the key being searched for;

search the node corresponding to the first symbol of the key for a pointer to a level of nodes having the logical level number as the key being searched for; and

search for an entry for the key on the level of nodes having the logical level number of the key being searched for.

15.     An article of manufacture, comprising:

a machine-readable medium comprising instructions, that when executed cause a machine to:

generate a tree for storing keys in a memory, the tree having a root node, each key having a string of symbols;

receive a key having an entry to be stored in a node of the tree; and

store an entry for the key in a node corresponding to the last symbol in the key in a level of nodes assigned a logical level number corresponding to the length of the key.

16.    The article of manufacture of claim 15, further comprising instructions, that when executed, cause a machine to:

a)    search for an entry for a key comprising a string of n symbols in a table of stored key entries organized as a tree of nodes by ascertaining a logical level number that corresponds to the length of the key;

b)    search a root node in the tree for a pointer to a node corresponding to a first symbol in the key;

c)    search the node corresponding to the first symbol in the key for a pointer to a next level of nodes;

d)    ascertain whether the next level of nodes is assigned the logical level number that corresponds to the length of the key;

e)    search for an entry for the key on the next level of nodes if the next level of nodes is assigned the logical level number;

f)    search for a pointer to subsequent next level of nodes if a parent level of nodes is not assigned the logical level number that corresponds to the length of the key; and

g)    repeat d-f until one of the key entry is found, there are no more pointers to subsequent levels of nodes, and a number of levels of nodes corresponding to a number of all the symbols in the key have been searched.

17.    The article of manufacture of claim 15, further comprising instructions, that when executed, cause a machine to:

follow a first pointer from the root node to the node corresponding to the first symbol in the key;

search the node corresponding to the first symbol in the key for a second pointer to a next level of nodes;

if the second pointer does not exist, then create a next level of nodes, storing an entry for the key in a node corresponding to the last symbol of the key, and assigning logical level number corresponding to the last symbol in the key to the level of nodes in which the key is stored;

22

if the second pointer exists and points to a level of nodes having a logical level number corresponding to a length of the key, then store an entry for the key in a node corresponding to the last symbol of the key; and

if the second pointer exists and points to a level of nodes not having a logical level number corresponding to the length of the key, then insert a new level of nodes to store an entry for the key, inserting the new level in sequence according to ascending logical level numbers of all the levels of nodes in the tree, the ascending starting at the root node.

23